Challenges to Building Scalable System Architectures

Keith D. Underwood

June 4, 2008

DEG
Architecture and Planning

Legal Disclaimers

Performance tests and ratings are measured using specific computer systems and/or components and reflect the approximate performance of Intel products as measured by those tests. Any difference in system hardware or software design or configuration may affect actual performance. Buyers should consult other sources of information to evaluate the performance of systems or components they are considering purchasing. For more information on performance tests and on the performance of Intel products, visit http://www.intel.com/performance/resources/limits.htm or call (U.S.) 1-800-628-8686 or 1-916-356-3104.

All dates and products specified are for planning purposes only and are subject to change without notice

Relative performance is calculated by assigning a baseline value of 1.0 to one benchmark result, and then dividing the actual benchmark result for the baseline platform into each of the specific benchmark results of each of the other platforms, and assigning them a relative performance number that correlates with the performance improvements reported.

SPEC, SPECint2000, SPECfp2000, SPECint2006, SPECfp2006, SPECjbb, SPECWeb are trademarks of the Standard Performance Evaluation Corporation. See http://www.spec.org for more information.

Intel® Virtualization Technology requires a computer system with an enabled Intel® processor, BIOS, virtual machine monitor (VMM) and, for some uses, certain platform software enabled for it. Functionality, performance or other benefits will vary depending on hardware and software configurations and may require a BIOS update. Software applications may not be compatible with all operating systems. Please check with your application vendor.

Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor series, not across different processor sequences. See http://www.intel.com/products/processor_number for details.

Intel products are not intended for use in medical, life saving, life sustaining, critical control or safety systems, or in nuclear facility applications. All dates and products specified are for planning purposes only and are subject to change without notice

* Other names and brands may be claimed as the property of others.

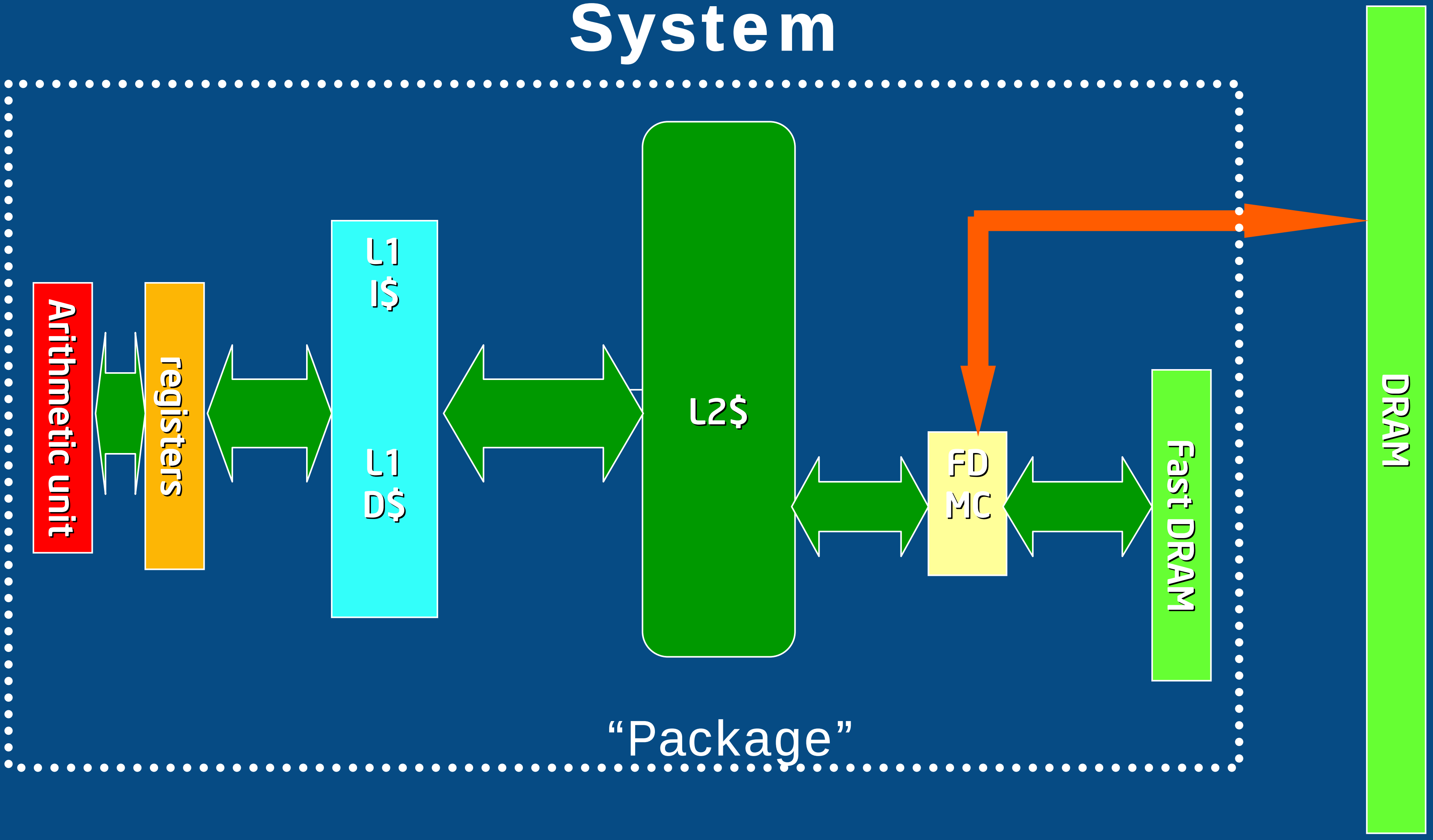
What is a Scalable System Architecture?

- A supercomputing center must be able to:
  - Buy it
  - Put it in a building
  - Turn it on (i.e. power and cool it)
  - Get the applications to run well on a node
  - Get the applications to scale
  - Run it for more than 15 minutes without failing
- Constraints that were second order have become first order
  - This will drive changes in architecture
  - Applications that want performance will have to come along for the ride
- Premise 1: You want an Exascale computer by ~2020
- Premise 2: You shouldn't stick your head in the sand for the next 5 years

Technology and Memory Accesses

- The energy per bit of data movement is decreasing very slowly
  - Remember, power is a first order constraint
  - Applications will have to explicitly manage (and minimize) data movement to enable “reasonable power” systems
- The cost per pin is decreasing VERY slowly while the bandwidth per pin is increasing VERY slowly
- All issues combine to suggest an alternative memory system
  - Explicit user management of a “large” local state
  - Significantly lower energy per bit
  - Significantly higher bandwidth / $$ by not using pins

Implications for the Memory System



Technology and Networks

- The power (mW/Gb/s) for bandwidth is only slowly decreasing
  - Very similar to memory problem
  - Not as easy of a fix
- The viable distance for electrical signaling is going down
- The power addition for optics is still too high
  - Near/medium term: 3D torus to minimize cost and power
  - Long term: Optics offer hope to break the topology constraint and mitigate the bandwidth constraint, but:
    - Power will remain a major challenge
- Note to applications: plan to optimize for locality!

Architecture and Networks

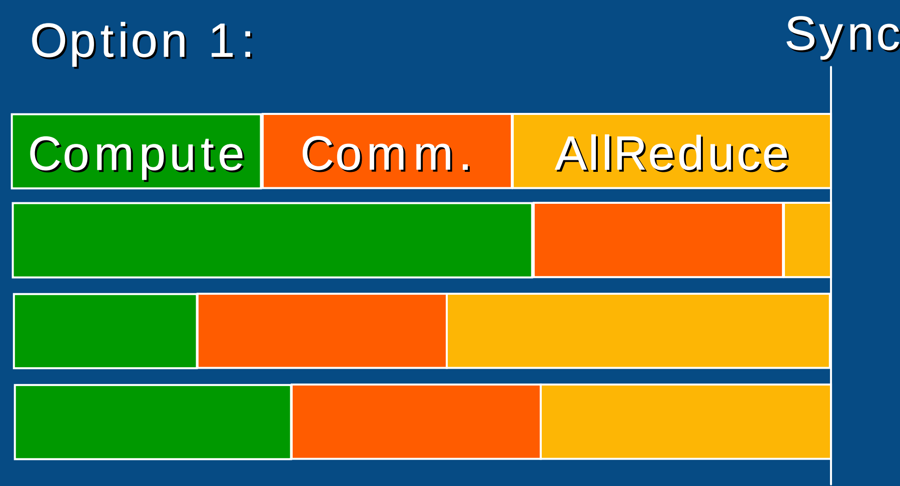
- Things a processor does badly
  – Walk linked lists
  – Extensive bit masking/comparisons/branches
  – i.e. things required for message header processing
- Do you really want a 90W processor spending its time doing things it does badly?
  – How many of you do visualization with a processor?
  – How many of you compute network checksums with a processor?
- The alternative: offload message processing to the NIC
  – Opportunity to dramatically improve message rate performance (don't use an embedded processor here)
  – Significantly lower power for message processing
  – Solve the semantic mismatch between MPI and the network

Processor Architecture will Impact NIC Architecture

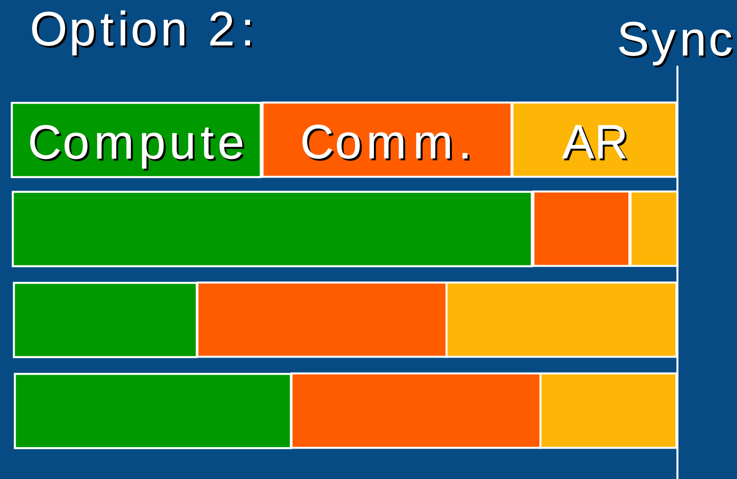
- Vast socket concurrency may pose challenges for the NIC
  - Resources (e.g. command queues)
  - Multiple concurrent network streams
- Some challenges require application answers
  - Must one thread get all of the bandwidth?
    - Pro: slightly mitigate load imbalance
    - Con: data path widths
  - Can you constrain MPI usage?
    - Pro: minimize requirements on NIC state
    - Con: additional constraints on the application

Bandwidth Allocation and Load Imbalance

Option 1:



Option 2:



An Exascale Future has "Simple" Cores

- "Magic" architectures are going to have to go away
- Processors currently "discover" concurrency and "predict" future work and data needs
    - Out-of-order computation
    - Branch prediction and large shared caches
- Users will have to express concurrency
- The future holds:
  - Smaller cores to get less average power per operation
  - User must "express" concurrency previously "discovered" by hardware
    - Substantially higher thread count per socket
    - Must find outstanding references to cover memory latency
    - More threads than cores to cover branch latencies

An Exascale Future is Homogeneous

- The RoadRunner* Experience
  - To date, every presentation I have seen of applications on Cell* have ended: “And, in the future, we will move the rest of the code to the SPE”
  - Why? Amdahl’s law.
- Power will drive a “lowest common denominator” approach
  - Cores *must* be optimized for average power per FLOP to achieve Exascale at rational power (100MW)
  - Special function units (e.g. encryption, compression, MPEG decoding) may exist, but are useless to HPC

Summary

- System power will become ***the*** constraint of the future for HPC computing
  - You better hope that it is for everything else
- Architectural innovations *can* provide substantial improvements
  - Explicitly managed, local memories can minimize cost of memory operations
  - Specialized hardware in the network can reduce power and increase performance
- An Exascale system will be homogeneous, but it can't use "big" cores
- Unfortunate result: The *programmer* must bear a lot more weight
  - Optimization for locality throughout the system
  - Explicit movement of data through the memory hierarchy

intel

Leap ahead™